Trigger Tables

Jonathan Lewis

Ace Training

Updated June 2002

Used for January 21, 2003 Training

What's in a Name

- **PHYSICS_1_01[10,71,301]**
  - **Table name**
    - **"1_01" is administrative major version**
  - **Table version**
    - **Name and version specify physics content**
  - **Level 2 tag set**
    - **CVS tags of Level 2 alpha code**
    - **Tied to physics table**
  - **Level 3 tag set**
    - **Level 3 executable, tcl and calibration set**
    - **Exe build driven from table**
    - **Code is base release plus a patch list**

Building and Testing

- **Table built with database GUI by trigger drones**
  - GUI instruction is on need-to-know basis
  - GUI performs consistency checks and builds L2 exe
    - Assigns L2 tagset
- **L3 gang builds tcl and exe**
  - Usually 1-2 hours
- **Initial test without beam**
  - This means no HEP colliding beam
  - Studies, injection, etc. are OK
- **Beam test usually at end of store**
  - Minimize integrated luminosity
- **If gurus approve, will make it default on the white board**

Building and Testing, 2

- Have ability to change L2 and L3 executables without new physics table
  - Fix bugs
  - Improve low-level code
- Occasionally will test new L2 or L3 tagset for existing table
- May subsequently change defaults
- For table (physics) changes, usually have round of PHYSICS_TEST tables before copying to PHYSICS
  - Attempt to limit confusion later

Decoupled Tables

- In usual mode of operation, Level 3 is driven by Level 2 decisions
  - Explicit paths
- For non-physics tables, can have Level 3 process all events the same way
  - Use for cosmics, L1/L2 tests, etc.
  - One tagset can be used for many tables
  - Usually have two current
    - Null
    - All reconstruction
  - Single output stream only
- Decoupled tables use different RunControl menu from physics tables